Did you evaluate your ontology? OOPS! (OntOlogy Pitfall Scanner!)

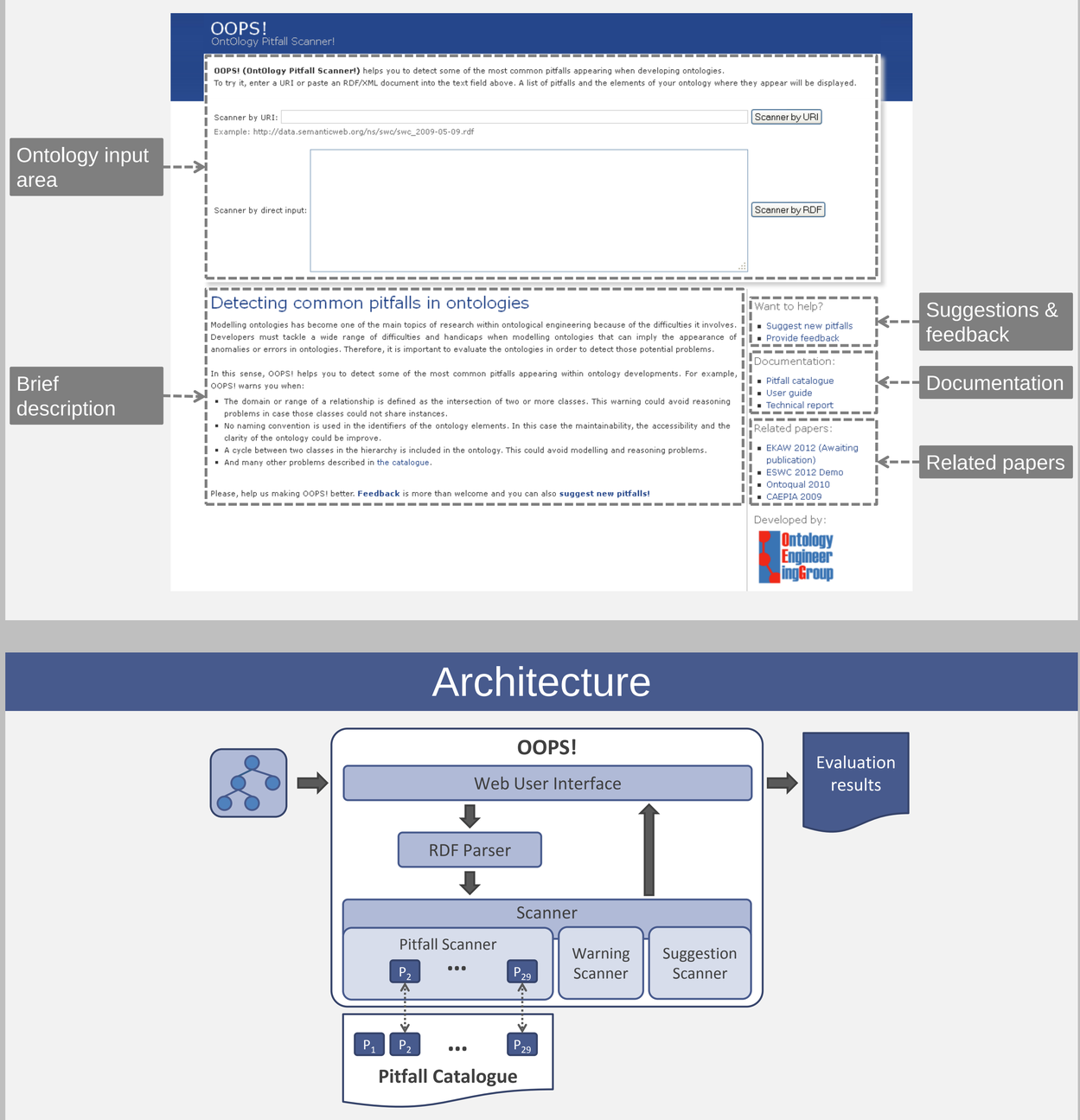
María Poveda-Villalón, Mari Carmen Suárez-Figueroa, Asunción Gómez-Pérez
Ontology Engineering Group (OEG). Universidad Politécnica de Madrid
{mpoveda, mcsuarez, asun}@fi.upm.es

Abstract

The application of methodologies for building ontologies can improve ontology quality. However, such quality is not guaranteed because of the difficulties involved in ontology modelling. These difficulties are related to the inclusion of anomalies or bad practices within the ontology development. In this context, our aim is to describe **OOPS! (OntOlogy Pitfall Scanner!)**, a tool for detecting pitfalls in ontologies.

Getting started

http://www.oeg-upm.net/oops

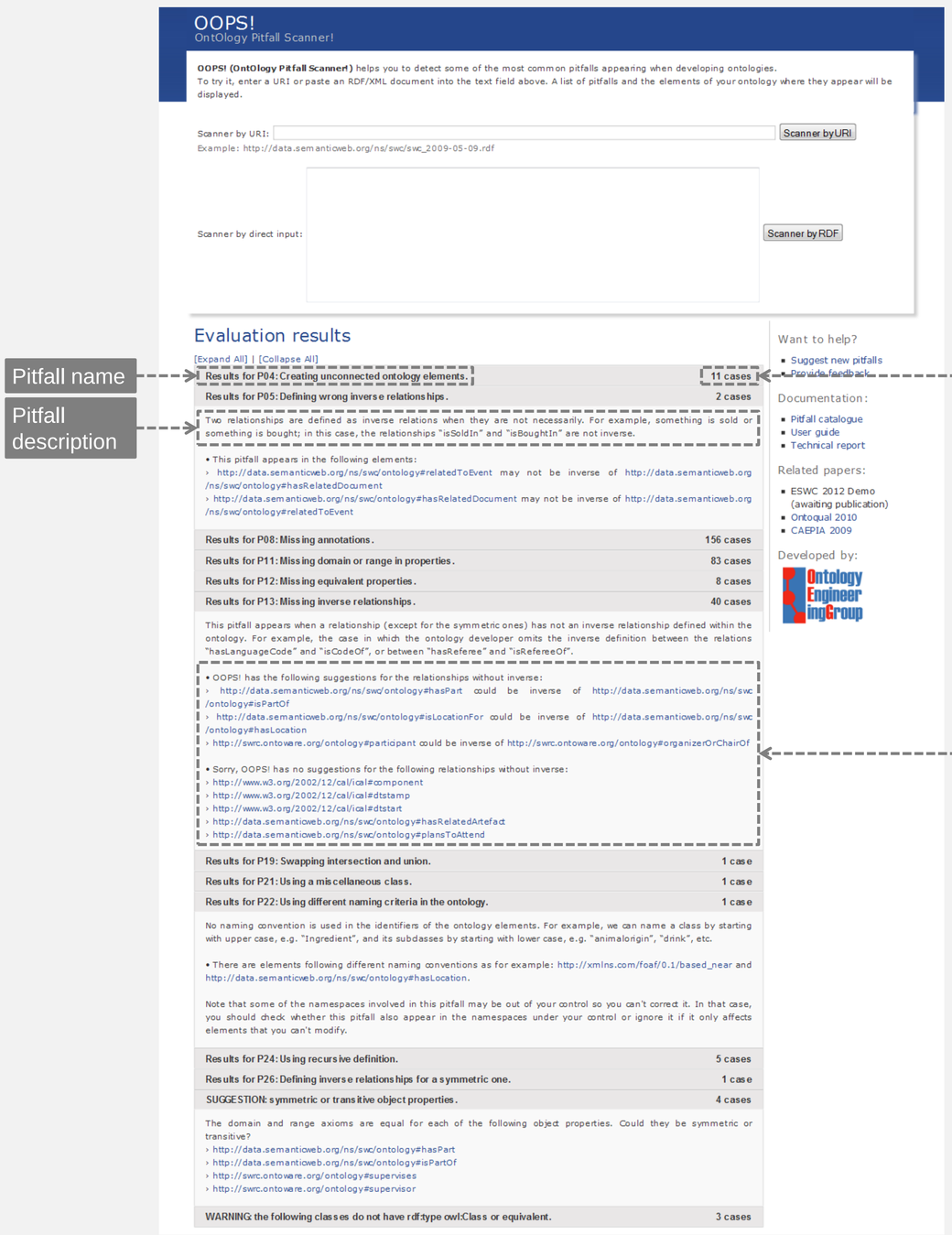


Architecture

Pitfall catalogue

P1. Creating polysemous elements
P2. Creating synonyms as classes
P3. Creating the relationship “is” instead of using "rdfs:subClassOf", "rdf:type" or "owl:sameAs"
P4. Creating unconnected ontology elements
P5. Defining wrong inverse relationships
P6. Including cycles in the hierarchy
P7. Merging different concepts in the same class
P8. Missing annotations
P9. Missing basic information
P10. Missing disjointness
P11. Missing domain or range in properties
P12. Missing equivalent properties
P13. Missing inverse relationships
P14. Misusing "owl:allValuesFrom"
P15. Misusing “not some” and “some not”
P16. Misusing primitive and defined classes
P17. Specializing too much a hierarchy
P18. Specifying too much the domain or the range
P19. Swapping intersection and union
P20. Misusing ontology annotations
P21. Using a miscellaneous class
P22. Using different naming criteria in the ontology
P23. Using incorrectly ontology elements
P24. Using recursive definition
P25. Defining a relationship inverse to itself
P26. Defining inverse relationships for a symmetric one
P27. Defining wrong equivalent relationships
P28. Defining wrong symmetric relationships
P29. Defining wrong transitive relationships

Scanning ontologies



Conclusions

- OOPS! represents a step forward within ontology evaluation tools as :
  a) it **enlarges the list of errors detected** by most recent and available works
  b) it is fully **independent** of any ontology **development environment**
  c) it works with **main web browsers** (Firefox, Chrome, Safari and IE)
- It is being **tested by OEG** members and **used by** ontology developers who belong to **different organizations** (such as AtoS, Tecnalia, Departament Arquitectura, La Salle at Universitat Ramon Llull, etc.).
- It is **freely available** to users on the Web. Everyone can test it, provide **feedback**, **suggest new pitfalls** to be included in the catalogue and implemented into the tool.

Next steps

- To enlarge the pitfall catalogue by including:
  - new errors that might be **detected** during our research
  - pitfalls **suggested by users**
- To **group** and **classify** pitfalls by categories and allow the user to **choose** categories or particular pitfalls to check the ontology against
  - OOPS! more flexible and adaptable to specific user needs
- To increase OOPS! features with **guidelines** about how to **solve** each **pitfall**
  - To facilitate the task of repairing the ontology after the diagnosis
- To associate **priority levels** to each pitfall according to their different types of consequences they can convey
  - To prioritize actions to be taken during the repairing task
- To make **REST services** available
  - To allow other developments to use and integrate the pitfall scanner functionalities within their applications
- To allow users to **define pitfalls** following a **formal language**, according with their particular quality criteria
  - To allow users to execute OOPS! in a customized way

Acknowledgments

This work has been partially supported by the Spanish projects BabelData (TIN2010-17550) and BuscaMedia (CENIT 2009-1026).

Posters & Demos Session at 18th International Conference on Knowledge Engineering and Knowledge Management (EKAW2012). 8 – 12 October, 2012. Galway City, Ireland.